

Stradus CMS v1.0 beta4

A könnyűsúlyú tartalomkezelő rendszer

(modulok írása)

2012-08-06



Tartalomjegyzék

Bevezetés.....	2
A modulok részei.....	3
A gedit beállítása.....	4
Modulok készítése, lehetőségek és beépített függvények.....	5
Hiperszöveges modul.....	5
PHP-s modul, globális lehetőségek.....	6
A simple_ml_tag_reader függvény.....	7
A simple_ml_tag_writer függvény.....	7
Globális AJAX függvény, AJAX hívás egy sorral.....	8
Menüpontok a modulon belül.....	9
Felhasználók és a modulok, hitelesítés.....	9
Karakterláncok ékeztelenítése.....	10
Modulok beállítófelülete.....	11
Kérjen segítséget!.....	13
Nyilatkozat a 'Stradus CMS' fejlesztőjétől.....	13

Bevezetés

Köszönjük, hogy a **Stradus tartalomkezelő rendszert** választotta.

A rendszer ún. modulokból építi fel a lapok tartalmát.

A modulok tulajdonképpen kész programkönyvtárak, amiknek ugyan vannak megkötöttségei, de bármilyen előre elkészített webprogram, könnyen és gyorsan átalakítható SCMS modullá. Azonban ha nincs mit átírni, újat készíteni sem bonyolult.



A Stradus tartalomkezelésről, a modulokról és a lapokról többet olvashat a Felhasználói Kézikönyvben, amit a <http://stradus.eu> webhelyről tölthet le.

A modulokat a *moduls* mappa tárolja a rendszer gyökérkönyvtárában.

Modulok írásához szüksége lesz:

- Helyi web- és adatbázisszerverre (a Stradus CMS futtatásához)
- FTP-kliensre (a rendszer és a modulok feltöltésére)
- Egy forrásszerkesztőre (WYSIWYG): getit, Notepad++, Bluefish
- Opcionálisan egy átalakítandó webprogramra, vagy egy saját ötletre
- HTML, PHP és MySQL ismeretekre
- Esetleg CSS, Javascript és AJAX ismeretekre



Ha kételkedik webfejlesztői tudásában keresse fel a <http://www.w3schools.com/> webhelyet!

A modulok részei

A rendszer a telepített modulok frissítése során két egyszerű szöveges fájlba menti a modulok listáit:

- *moduls/level1_moduls*: Az 1-es szintű felhasználók (tulajdonosok) számára menedzselhető modulok listáját tartalmazza. A modulnevek felkiáltójellel vannak elválasztva egymástól.
- *moduls/valid_moduls*: Tartalmazza az összes, frissítéskor érvényesnek ítélt modulnak a nevét, felkiáltójellel elválasztva.

A modulok egy ellenőrzésen mennek keresztül frissítéskor, mert meg kell felelniük bizonyos formai elvárásoknak és vannak kötelező elemei a saját mappáján belül:

- **editor.php**: A modul beállítófelületét tartalmazza. A rendszer nyújt némi segítséget az elkészítéséhez. Egy következő fejezetben részletesen olvashat róla.
- **info.xml**: A modul információs fájlja. Ez az egyszerű XML fájl tárolja a modul egyéb beállításait és adatait:
 - **name**: A modul neve
 - **path_name**: A modul mappájának a neve. Szükséges lehet a későbbi program megfelelő működéséhez.
 - **author**: A modul szerzőjének neve
 - **version**: A modul verziószáma
 - **db_field_names**: Minden modulnak létrejön egy táblája az adatbázisban. Ennek a táblának a tetszőleges mezői, vesszővel elválasztva. A tábla modulfrissítéskor jön létre. Minden modulegyedhez egy sor tartozik ebben a táblában.
 - **include**: A moduloknak két különböző üzemmódja lehet. Ha az *include* tag tartalma **0**, akkor csupán egyszerű szöveges sablonokat lehet készíteni egy modulból (hiperszöveges modul), azonban ha **1**, akkor a *model.php* egyből beillesztődik a tartalomba (PHP-s modul).
 - **content**: A modul mappájának felsorolásszerű listáját jelenti. Csupán tájékoztató jellegű.
 - **level1**: Ez határozza meg, hogy az 1-es szintű felhasználók is láthassák és szerkeszthessék a modult és annak egyedeit. Beállítása szintén a logikai **1**-el és **0**-val történik.
 - **unique**: Ez a tag még nincsen használatban. Előfordulhat, hogy egy modulnak csak egy egyedet ajánlott létrehozni. Ezzel a tag-gel lehet majd letiltani az egyedek létrehozását a beta5-ös verziótól.

- **model.php**: Ez a PHP fájl tartalmazza a modulegyedek tényleges felületét. Ez a fájl fog megjelenni a tartalomban, miután a modul egy egyedét beillesztette egy lapra. Egy következő fejezetben részletesen olvashat róla.
- **A telepített modulok táblái**: A modulok táblái a modulfrissítéskor jönnek létre. Minden egyes rekord, egy modulegyedhez tartozik. Az *info.xml*, *db_field_names* értéke határozza meg a tetszőleges mezőket.

Telepített modul táblája
id: smallint(6), NOT NULL, AUTO_INCREMENT
name: varchar(50), DEFAULT: 'New *modul_neve*'
tetszoleges_mezo: text, DEFAULT: NULL
tetszoleges_mezo: text, DEFAULT: NULL
...

A gedit beállítása

A gedit, ami többek között az Ubuntu operációs rendszer alapfelszereltsége, gyárilag egy egyszerű szövegszerkesztőnek tűnik, de könnyedén beállítható, hogy a forráskódok átláthatóbbak legyenek.

1. A menüsávon navigáljon a *Szerkesztés* → *Beállítások* menüpontra.
2. A *Nézet* fülön pipálja be a következőket:
 - Sorok számának megjelenítése
 - Jelenlegi sor kiemelése
 - Zároljelpárok kiemelése
3. A *Betűk és színek* fülön válasszon egy tetszőleges színsémát. A *Kobalt* színösszeállítás ajánlott – kíméli a szemet.

Modulok készítése, lehetőségek és beépített függvények

Az első lépés során el kell készíteni az *editor.php*, *info.xml* és *model.php* fájlokat.

A két PHP fájl tartalmára vonatkozóan nincsenek megkötöttségek, de az *info.xml*-t ajánlott a következő alapján létrehozni:

```
<?xml version="1.0"?>
<info>
<name>Bármilyen modul</name>
<path_name>barmilyen_modul</path_name>
<author>Stradus CMS</author>
<version>1.0</version>
<db_field_names>első_mezo,második_mezo</db_field_names>
<include>0/1</include>
<content>info.xml,model.php,editor.php</content>
<level1>0/1</level1>
<unique>0/1</unique>
</info>
```

Hiperszöveges modul

A moduloknak két különböző üzemmódja lehet. Ha az *include* tag tartalma az *info.xml* fájlban 0, akkor csupán egyszerű hiperszöveges sablonokat lehet készíteni egy modulból.

Elsőre talán nem tűnik túl hasznos lehetőségnek, mert aránylag kevés esetben használható pont a korlátozottsága miatt, de a beépített *simply_text* modul is ezen az elven működik.

Ez a módszer egyszerű behelyettesítéssel „cseréli” a tartalmat.

A *model.php* gyakorlatilag nem is PHP programot, hanem csak egy egyszerű HTML „modell” tartalmaz.

Lássuk a *simply_text model.php*-jét:

```
<div style="text-align:##align##;margin:13px;">
<font size="##size##">##text##</font>
</div>
```

A dupla hashmark közötti pozíciókba fognak kerülni az egyed táblájába tetszőlegesen elkészített mezők értékei. Egyszóval ha...

```
<db_field_names>valami,igazitas</db_field_names>
```

...az *info.xml*-ben, akkor egy...

```
<p align="##igazitas##">##valami##</p>
```

...tartalmú modellel könnyedén lekérdezhetjük a valami és igazitas mezőt.

Így, ha az adatbázisban valami="Hello do" és igazitas="right", akkor a kimenet:

```
<p align="right">Hello do</p>
```

Így könnyedén csinálhatunk ugyanolyan felépítésű, de különböző tartalmú egyedeket, de dinamizmusra, vizsgálatokra ezzel nincs lehetőség.

PHP-s modul, globális lehetőségek

A PHP-s modulok lehetőségeinek a tárháza tulajdonképpen kifogyhatatlan. Ha elkészítette a modulok fájljait és az *include* értéket 1-re állította, egy ellenőrzés után már kezdheti is a programírást.

Tekintse meg a *simply_image_model.php* forrását:

```
<?php
print '<div style="margin:20px;';
    if($field['align'] == 'left' || $field['align'] == 'right')
    {
        print 'float:'. $field['align'].';';
    }

    if($field['align'] == 'center')
    {
        print 'text-align:'. $field['align'].';';
    }
print '">

</div>';
?>
```

Látszik, hogy egy mezei PHP programról van szó, annyi kis különbséggel, hogy már vannak alapesetben a rendelkezésre bocsátott globális változók és pár hasznos beépített függvény, amit a Stradus CMS bocsát rendelkezésre.

A *\$field* tömbben érhetőek el az egyedhez tartozó adatbázismezők értékei. Tehát ha...

```
<db_field_names>valami,igazitas</db_field_names>
```

akkor a

```
$field['id']
$field['name']
$field['valami']
$field['igazitas']
```

...változók elérhetőek lesznek. Az *id* a kérdéses egyed sorszámát, a *name* a nevét tartalmazza.



Az egyedek tetszőleges adatbázismezői mindig text típusúak lesznek!

A simple_ml_tag_reader függvény

Ha esetleg a modul info.xml fájljából szeretnénk olvasni mezőket, használjuk a

```
simple_ml_tag_reader('ml_fajl_utvonala/fajl.xml', 'tag')
```

függvényt.

Ez a függvény első paraméterének egy elérési utat kell megadni, pl. a modul információs fájljának útvonalát, a másodiknak meg a kívánt taget, és a nyitó és záró tagpárok közötti részt adja vissza. Ez a függvény csak xml-szerű fájlkból tud sikeresen olvasni, ahol a tagek nem ismétlik egymást. Amennyiben ismétlés van, az első megtalált tag tartalmát adja vissza.

A simple_ml_tag_writer függvény

Ez a beépített függvény, a simple_ml_tag_reader() testvére. Arra való, hogy módosítsuk az olyan egyszerű felépítésű xml fájlok tartalmát, amiket az SCMS is használ.

Három paramétert vár bemenetűl:

1. Az XML fájl elérési útvonala
2. A módosítani kívánt tag
3. String, ami a tagpár új tartalma lesz

```
simple_ml_tag_writer('ml_fajl_utvonala/fajl.xml', 'tag', 'bármi')
```


Globális AJAX függvény, AJAX hívás egy sorral

Az SCMS rendelkezik beépített AJAX-hívó JavaScript függvénnyel is, amely kifejezetten tökéletes például modulok működtetésére, de a rendszer is, főleg az adminisztráció során, rengeteget használja, de az AJAX-os lapváltó mód is ezzel működik.

A `global_ajax()` hat paramétert vár bemenetétül:

1. A meghívni kívánt fájl (*string*)
2. A változónevek tömbje (*array*)
3. A változónevekhez tartozó értékek tömbje (*array*)
4. Eldöntheti, hogy a `global_ajax()` a meghívott fájl válaszát írja ki az 5. paraméterben megadott helyre, vagy a 6. paraméterben megadott stringet. (*boolean*)
5. Azt az ID-t tartalmazza, aminek az innerHTML-jébe kiíródjon a meghívott fájl válasza, ha a 4. paraméter *true*. A függvény a `document.getElementById('azonosito').innerHTML` utasítást használja. Ha nem kíván élni a kiíratás lehetőségével, helytöltőnek hagyjon egy üres karakterláncot: " (*string*)
6. Az üzenet amit a függvény visszaadon, ha a 4. paraméter értéke *false*. Ha nem kíván élni a kiíratás lehetőségével, helytöltőnek hagyjon egy üres karakterláncot: " (*string*)

A `global_ajax()` *true* vagy *false* logikai értékeket ad vissza, attól függően, hogy az AJAX hívás sikeres volt-e. Bármilyen hiba merül fel, a visszaadott érték *false* lesz.

A `global_ajax()`-ot általában két tömb társaságában használhatja, jellemzően ebben a formában:

```
varnames = ["valtozo_nev1", "valtozo_nev2", "valtozo_nevn"];
values = [1,2,n];
global_ajax('meghivott.php', varnames, params, true, 'div_id', '');
```

Így a `meghivott.php` válasza a `div_id` azonosítójú konténer innerHTML-jébe íródik ki.



A lapváltó módokról (PHP-s és AJAX-os) többet olvashat a felhasználói kézikönyvben, amit letölthet a <http://stradus.eu> webhelyről!

Menüpontok a modulon belül

Ahhoz, hogy navigálhatni lehessen úgymond egy „modulon belül”, szóval ha egy modul gyakorlatilag külön menüpontokra ágazik, több lehetőség adódik.

- A **global_ajax** függvénnyel, vagy egy egyedi kóddal elkészített hívással cserélheti egy div tartalmát.
- Használhatja a kevésbé elegáns iframe-es megoldást.
- Kíráthatja az összes modulon belüli menüpont tartalmát egy oldalra, és a CSS **display:none**; paranccsal láthatatlanná teheti a nem kívánt tartalmat, majd a **document.getElementById('azonosito').style.display = inline** (vagy block) JavaScript utasítással láthatóvá teheti azokat.
- Felépítheti PHP-s elágazásokból a menüpont váltást a modul modelljében, de ehhez fog kelleni egy apróság.
Ha a Stradus CMS-t PHP-s lapváltó üzemmódban használja, a **\$url_support**, úgymond rendszeren belül globálisan, tartalmazza azt az URL-t, amin éppen a rendszer áll azaz, hogy melyik lapon tartózkodik a felhasználó. Ehhez a stringhez hozzáfűzheti az átadni kívánt paramétereket, hogy a modulja fel tudja azokat dolgozni. Amennyiben nem használja ezt a lehetőséget, lehetséges, hogy a böngésző nem arra a lapra fog érkezni, amin a modulja megtalálható, és így a modulon belüli értékátadás nem hajtodik végre.
Ez a lehetőség csak a rendszer PHP-s lapváltó üzemmódjában működik megfelelően!

Felhasználók és a modulok, hitelesítés

Könnyen előfordulhat, pl. egy fórummotor írása közben, hogy szüksége lesz a belépett felhasználó adataira.

A Stradus CMS sűtikben tárolja a belépett felhasználó hitelesítő adatait. (A sűtik session-nel jönnek létre, így kevésbé kell félni a jelszólopástól persze egy kis elővigyázatosság sosem árt.)

Ezek a sűtik (session változók) így is elérhetőek:

```
$_COOKIE['whosloggedin'] //Felhasználónév  
$_COOKIE['userspassword'] //Jelszó  
$_COOKIE['userslevel'] //Felhasználói szint
```



A felhasználói szintekről többet olvashat a felhasználói kézikönyvben, amit letölthet a <http://stradus.eu> webhelyről!

Egy fórum esetében jól jöhet, ha egy 1-es szintű felhasználó (fórum működés/beállítástól függően) nem látja az „Új topik létrehozása” gombot, hanem csak az adminisztrátoroknak jelenik meg, sőt csak ők tudjanak létrehozni. Ezt egy egyszerű levizsgálással ezt elintézhettük.

Hogy egy bejelentkezett felhasználó hitelességéről megbizonyosodhasson, a rendszer rendelkezésre bocsátja az authentication() függvényt, ami nem vár paramétert.

A felhasználói szinttel és kapcsolatban vizsgálva, biztosak lehetünk benne, hogy egy hiteles adminisztrátor van bejelentkezve.

```
If($_COOKIE['userslevel']>1 && authentication()==1)
{
print 'Egy hiteles admin van bejelentkezve!';
}
```

Látszik, hogy az authentication() 1-et ad vissza akkor, ha a felhasználói session változók hiteles adatokat tartalmaznak. Ha nem, akkor a visszaadott érték 0 lesz.

Karakterláncok ékeztelenítése

Kevésbé fontos, ám előfordulhat, hogy hasznos beépített függvény lehet az accent_remover(), ami egyetlen bemenetének egy karakterláncot vár, és annak ékeztelenített változatát adja vissza.

A függvény a következő karaktereket cseréli le:

ö, ü, ó, ő, ú, ű, é, á, í valamint a szóközt

A felsorolt karaktereket a következőkre cseréli:

o, u, o, o, u, u, e, a, i, _

Látszik, hogy a szóközből, alsó vonás lesz.

Használható ez a függvény pl. az átlag felhasználók által feltöltött fájlok nevének átalakítására!

Modulok beállítófelülete

A beállítófelületet a modulegyedek testreszabására használhatjuk. A moduloknak, az egy egyedhez tartozó tetszőleges adatbázis mezőit szerkeszthetjük vele, de egyéb módszerekkel (AJAX és iframe), még gazdagabb lehetőségeket hozhatunk létre.

Készíthet modulokat beállítófelület nélkül is, csak az *editor.php* fájl megléte fontos. Ha nem szeretné a modult később állítgatni, mert pl. csak saját felhasználásra készül, és megcsinálja olyannak egyből, amilyennek szeretné, elég csak üresen hagyni az előbb említett fájlt.

A beállítófelület is egy szabványos PHP fájl, de van egy apró megkötöttsége a rendszer működéséből adódóan: Nem tölthet újra a lap. Mivel a beállítófelületet a rendszer AJAX-szal hívja meg, egy esetleges újratöltés elronthatja a munkafolyamatot, és az addigi beállítások is elvesznek.

Úgy ahogy a modell esetében itt is rendelkezésre állnak az egyedekhez kapcsolódó globális változók.

Tehát ha...

```
<db_field_names>valami,igazitas</db_field_names>
```

akkor a

```
$row['id']  
$row['name']  
$row['valami']  
$row['igazitas']
```

...változók elérhetőek lesznek. Az *id* a kérdéses egyed sorszámát, a *name* a nevét tartalmazza.



**A \$row nem keverendő össze a model.php \$field tömbjével!
Nem helyettesíti egyik a másikat.**

Már az előző oldalakon bemutattuk a *simple_image* modul modelljét, azonban a *editor.php*-je sem sokkal bonyolultabb:

```
<?php  
print '<p>Képfájl<br />  
<select name="image">';  
$dirname = "../modulok/simple_image/kepek";  
$dir = opendir($dirname);  
while(!is_bool($file=readdir($dir)))  
{  
    if(!is_dir("$dirname/$file"))
```

```

    {
    print '<option';
    if($row['image']=='modulok/simply_image/kepek/'.$file){print
' selected="selected"';} print '
value="modulok/simply_image/kepek/'.$file.'">'.$file.'</option>';
    }
}
print '
</select> </p>
<p> Igazítás<br />
<input type="radio" name="align"; if($row['align']=='left')
{print ' checked="checked"';} print ' value="left">Körülfolyatás
jobbról (balra)</input>
<input type="radio" name="align"; if($row['align']=='center')
{print ' checked="checked"';} print '
value="center">Középre</input>
<input type="radio" name="align"; if($row['align']=='right')
{print ' checked="checked"';} print ' value="right">Körülfolyatás
balról (jobbra)</input>
</p>
<p>Szélesség<br /><input value="'.$row['size'].'" name="size"
/>px</p>
<p>
Képfeltöltés<br />
<iframe frameborder="0" src="modulok/simply_image/upload.php"
width="100%" height="170px"></iframe></p> '
?>

```

Nem szeretnénk a webprogramozói alapismeretek taglalásába bocsátkozni, ezért röviden részletezzük a forráskódot.

- Az elején egy select beviteli mezőbe, egy ciklussal beolvassuk a modulba feltöltött képeket. Mivel ez egy egyszerű képmegjelenítő modul, a beállítófelületen valamivel ki kell választani a megjeleníteni kívánt képet. Közben egy egyszerű levizsgálással egy option taghez hozzáfűzzük a selected argumentumot, ahol épp szükséges. Ez utóbbinál vizsgáljuk a `$row['image']` tömbelemet, ami a feltöltött kép elérési útját tartalmazza.
- Majd a választógomboknál, ahol az igazítást állítjuk, a `$row['align']` változót vizsgáljuk a feltételben.
- A szélességnél, ami egy egyszerű beviteli mező, egyszerűen beolvassuk a value argumentum értékének a `$row['size']`-ot.]
- Majd, hogy az újratöltést elkerüljük, a képfeltöltő-form egy iframe-ben helyezkedik el.

A *simply_image*-hez viszonyítva már el lehet indulni, egy saját modul fejlesztésének az irányába.

Kérjen segítséget!

Az SCMS-hez már készült termékkatalógus-kezelő, e-mail küldő és képgaléria. Tulajdonképpen bármi megvalósítható, csak be kell tartani azt a kevés szabályt amit a rendszer kíván és élni a lehetőségeivel.

- További információkért látogasson el a <http://stradus.eu> webhelyre!
- Kérdezhet a stradus@sznet.hu e-mail címen!
- Ha még nem tette, olvassa el a felhasználói kézikönyvet, ami a webhelyünkről letölthető!
- Amennyiben hibabejelentés szeretne tenni, azt is megteheti a fenti lehetőségeken keresztül, vagy a rendszerbe épített hibajelentő szolgáltatáson keresztül.

Nyilatkozat a 'Stradus CMS' fejlesztőjétől

A Stradus webes tartalomkezelő-rendszer, csak teljes egészében, szabadon terjeszthető, illetve bármely része módosítható, amennyiben a módosult SCMS változat névjegyében feltüntetik annak eredetét, azaz a 'Stradus CMS' logóját, weboldalát, valamint supportjának elérhetőségét. Bármilyen módosított változatról kérem értesítsenek e-mailben.

A rendszer csak jogtisztá, nagyrészt nyílt forráskódú szoftverek felhasználásával készült.

Ön a rendszer telepítésével elfogadja a Stradus Végfelhasználói Licencszerződést. Kérem olvassa el figyelmesen

Sárfi Benjámín - 2012.08.05