

## CodeIgniter framework

Brezovský Michal · Informačné technológie, Študentské práce

06.01.2012



CodeIgniter je výkonný open source PHP framework. Bol primárne určený pre programátorov, ktorí potrebujú jednoduchú a elegantnú sadu nástrojov na vytvorenie plne funkčných webových aplikácií. Podobne ako iné frameworky, aj CodeIgniter si berie za úlohu umožniť vývojárom rozvoj projektov oveľa rýchlejšie oproti písaniu kódu od nuly, preto v sebe zahŕňa bohatú sadu knižníc pre bežné použitie. Tento framework je vyvíjaný spoločnosťou, a nie komunitou ako je tomu u iných frameworkov. Prvá verzia bola vydaná 28. februára 2006 spoločnosťou EllisLab, Inc.. Posledná vydaná verzia je 2.0.2.

### Základné vlastnosti

- codeigniter je zadarmo, distribuovaný pod licenciou Apache/BSD
- codeigniter je malý, systém vyžaduje iba základné knižnice. Všetky dodatočné knižnice sa volajú dynamicky na vyžiadanie
- codeigniter je rýchly a dosahuje skvelý výkon v porovnaní s inými frameworkami
- codeigniter používa MVC vývojový vzor, ktorá umožňuje oddeliť aplikačnú a riadiacu logiku od prezentačnej vrstvy
- codeigniter je rozširiteľný pomocou vlastných knižníc, "helperov" a rozšírení tried
- codeigniter podporuje coolURI's, nevyžaduje dodatočný template system, je dôkladne zdokumentovaný, má komunitné fórum s množstvom knižníc, helperov, rozšírení...

Podpora databáz: MySQL (4.1+), MSSQL, PostgreSQL, Oracle, SQLite, ODBC

### Inštalácia

Pred inštaláciou je potrebné stiahnuť zdrojové súbory z oficiálnej stránky <http://codeigniter.com/downloads/>.

1. rozbaľiť .zip archív
2. nahrať súbory na server.
3. otvoriť súbor application/config/config.php a nastaviť základnú URL adresu servera. V prípade použitia šifrovania je potrebné nastaviť šifrovací kľuč.
4. databázové prihlasovacie údaje sa nastavujú v súbore application/config/database.php

### Model-View-Controller

Codeigniter je založený na MVC vývojovom vzore. MVC je softvérový prístup, ktorý oddeľuje aplikačnú logiku od prezentačnej. V praxi to umožňuje lepšie využiť napísaný kód, vnáša to do aplikácie viac prehľadnosti a umožňuje pracovať na projektoch viacerím členom tímu, pričom každý člen pracuje na svojej úlohe nezávisle (backend programátori, dizajnéri, tester...).

Model reprezentuje dátové štruktúry, typicky model obsahuje funkcie ktoré umožňujú vkladať, vyberať, upravovať a mazať dáta nad databázou. View reprezentuje informácie, ktoré sú predkladané užívateľovi. Vo väčšine prípadov ide o HTML šablónu, do ktorej sú pri vykresľovaní nalievajúce dáta. Ale view môže byť aj PDF šablóna alebo email šablóna. Taktiež je možné do codeignitera implementovať niektorý z dostupných šablónovacích systémov (napr. smarty). Controller slúži ako prostredník medzi View a Model, obsluhuje požiadavky užívateľa a spája údaje z modelu a posiela ich do view. Codeigniter má voľnejší prístup k MVC ako iné framework-y. Ak v určitých prípadoch model nie je potrebný, je možné ho úplne vynechať a aplikáciu vyvíjať bez neho.

## **Ďalšie špecifické časti frameworku**

### **Helper**

Ako názov napovedá ide o pomocníkov pri práci, každý helper je zbierka funkcií ktoré sú si tematicky blízke. Codeigniter helper-i nie sú písané objektovo, ide o jednoduché procedurálne funkcie, pričom každá funkcia vykonáva svoju úlohu nezávisle na inej funkcii.

Ak nie je helper defaultne načítaný systémom je potrebné ho načítať manuálne z kódu kde bude volaný. Codeigniter podporuje aj vytváranie vlastných helperov, tie sú uložené v ceste application/helpers kde sú hľadané prioritne, ak tam nie sú nájdené, tak potom sa hľadajú v ceste system/helpers. Týmto spôsobom je možné "preťažiť" funkcie zo systémových helperov. "Preťažiť" je možné buď celý helper, alebo len niektorú z funkcií.

### **Knižnice**

Knižnicami sa myslia triedy uvedené v zozname tried v dokumentácii Codeigniter-a. V tomto prípade však ide o vlastné knižnice ktoré sú umiestnené v ceste application/libraries. Vlastné knižnice umožňujú:

- vytvoriť úplne nové knižnice
- rozšíriť natívne knižnice
- nahradiť natívne knižnice

### **Ovládače**

Ovládače sú špeciálny typ knižnice, ktoré majú rodičovskú triedu a ľubovoľný počet detských tried. Detské triedy majú prístup k rodičovskej triede, ale nie k súrodeneckej. Ovládače poskytujú elegantnú syntax v controlleroch pre knižnice.

### **Auto-load**

Auto-load v architektúre codeigniter-a predstavuje zavádzač ktorý zabezpečuje načítanie knižníc, helperov, modelov, ktoré budú inicializované pri každom spustení systému.

## Routes

Pomocou súboru v ceste application/config/routes.php je možné vytvárať COOL URI'S. Štruktúra súboru je nasledovná:

```
$route['product/{:inum}'] = "catalog/product_lookup_by_id/{:id}";
```

kde v prvej časti je vzor URL adresy z prehliadača, pričom vzory podporujú aj regulárne výrazy. Druhá časť je namapovanie URL adresy na konkrétny controller v rámci aplikácie, konkrétnu metódu a dynamický parameter v tomto prípade predstavuje premennú ku ktorej sa zvnútra controllera pristupuje nasledovne:

```
$this->uri->rsegment(1);
```

## Bezpečnosť

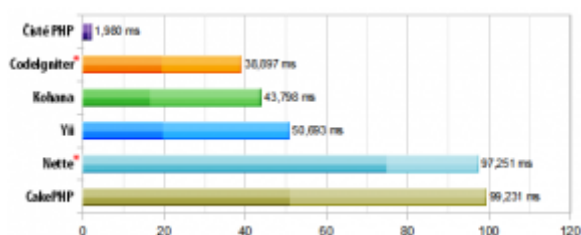
URI bezpečnosť: codeigniter bol navrhnutý tak aby dokázal minimalizovať prienik nebezpečných znakov v URI reťazci. Využíva helper na kontrolu validity vstupných parametrov. Globálne premenné: počas systémovej inicializácie sú všetky globálne premenné odstavené s výnimkou tých ktoré sú nájdené v poliach \$\_GET, \$\_POST, \$\_COOKIE. Error reporting: codeigniter rozlišuje produkčný a vývojový mód, pri zapnutí produkčného módu je akýkoľvek chybový výstup do prehliadača užívateľa potlačený. Naopak vo vývojovom režime je chybový výstup s detailnými informáciami zasielaný do prehliadača programátora. Pri prijatí akýchkoľvek dát sa odporúča:

- filtrovať dáta
- overovať dáta tak aby mali korektný typ, dĺžku, veľkosť...
- escape-ovať dáta pred odoslaním do databázy

Na tieto operácie poskytuje codeigniter účinné nástroje.

## Výkon

Ako už bolo spomenuté, Codeigniter dosahuje skvelý výkon, pre porovnanie uvádzame test niekoľkých PHP frameworkov. Súčasťou testu je porovnanie výkonu oproti čistému PHP. Uvedený test je postavený na veľmi jednoduchom MVC, v modeli sa načíta 10 riadkov z MySQL databázy, ktoré sa vypíšu vo view. Z výsledkov testu vyplýva, že codeigniter skutočne dosahuje skvelý výkon aj v porovnaní s inými bežne používanými PHP frameworkami.



---

## Charakteristika niektorých balíkov

### Session

Session umožňuje udržať si údaje užívateľa ktoré sa vzťahujú k jeho aktuálnemu sedeniu aj medzi viacerými "preklikmi" stránky. Session v codeigniter-i serializuje dáta do cookies s voliteľným šifrovaním, prípadne je možné session údaje serializovať aj do databázy, a do cookies sa bude ukladať len identifikátor. Pri povolenom šifrovaní je aj identifikátor zašifrovaný a takýto prístup nepredstavuje tak žiadne bezpečnostné riziko. Za zmienku stojí užitočná funkcia flashdata, ktorá umožňuje po zápise jednorazové využitie údajov. Po prístupe k nim sú tieto údaje automaticky odstránené so session.

### Calendar

Tento balík umožňuje dynamicky vytvárať kalendáre. Je možné naformátovať každý jeden aspekt kalendára.

### Shopping cart

Tento balík je vhodný najmä pre riešenia orientované na nákup cez internet. Táto trieda pridáva položky do session, tomu užívateľovi ktorý zostáva aktívni. Ďalej poskytuje metódy pre aktualizáciu, odstraňovanie a výpis záznamov.

### Image manipulation

Tento balík poskytuje rozhranie pre manipuláciu s grafickým obsahom. Podporované PHP knižnice sú GD/GD2, NetPBM, ImageMagick. Balík podporuje tieto funkcie pre manipuláciu:

- zmena veľkosti obrazu
- vytváranie náhľadov
- orezávanie obrazu
- rotácia obrazu
- vytváranie vodoznaku

O ďalších balíkoch je možné sa dozvedieť viac na adrese [http://codeigniter.com/user\\_guide](http://codeigniter.com/user_guide).

### Ukážka zdrojových kódov

Na ukážke je naznačené členenie view. V tomto prípade sa jedná o kombináciu XHTML kódu s PHP časťami. Uvedený kód zabezpečuje vytvorenie formuláru a tagov input pomocou vstavaných helper-ov.

```

<h1><?php echo $heading;?></h1>
<? echo form_open(
    'users/editsend/'.$row['ID'].'/'.$row['id_member']
);
?>
<table cellspacing = "10" border="1" cellpadding = "5">
<tr>
<td>idno: </td>
<td><?php echo form_input(array(
    'name' => 'name',
    'value' => $row['name'],
    'maxlength' => '30')); ?>
    <?=$this->validation->name_error; ?>
</td>
</tr>

```

Na ukážke je možné vidieť spracovanie odoslaného formuláru v príslušnom controlleri. Najskôr sa načítajú údaje z formuláru pomocou zabudovaného helperu input, ktorý pomocou metódy post vyžiada post data z formuláru. Potom sa pomocou knižnice validation zdefinujú pravidlá pre jednotlivé polia, a vykoná sa validácia. Ak sú vstupy z formulára vyhovujúce tak sa zavolá metóda z modelu.

```

function editsend($id, $m_id)
{
    $data = array();
    $data['name'] = $this->input->post('name');
    $data['surname'] = $this->input->post('surname');
    $data['phone'] = $this->input->post('phone');
    $data['email'] = $this->input->post('email');

    $this->load->library('validation');
    $rules['name'] = "required";
    $rules['surname'] = "required";
    $rules['phone'] = "required";
    $rules['email'] = "trim|required|valid_email";
    $this->validation->set_rules($rules);
    $this->validation->set_fields($data);

    if ($this->validation->run())
    {
        $this->load->model('Users_model', 'model_users');
        $this->model_users->Edit($id, $data);
    }
}

```

Na ukážke je vytvorenie modelu, s metódou edit ktorá vykonáva editovanie užívateľov v databáze. Využíva sa zabudovaná trieda DB, ktorá poskytuje veľmi rýchlu abstraktnú vrstvu s bežnými štruktúrami a active records vzormi.

```

class Users_model extends Model {

    function Users_model() {
        parent::Model();
    }

    function Edit($id, $data) {
        $this->db->where('ID', $id);
        $result = $this->db->update('users', $data);
        return (bool) $result;
    }
}

```

## Záver

Vďaka jednoduchému a logickému spracovaniu, ako aj prehľadne napísanej dokumentácií je Codeigniter vhodný predovšetkým pre začínajúcich programátorov, ktorí ešte nemajú bohaté skúsenosti s frameworkami. Rýchle spracovanie požiadaviek dovoľuje použiť tento framework aj na väčšie projekty.

