

## Zend Framework

Aust Peter · Informačné technológie

17.02.2012

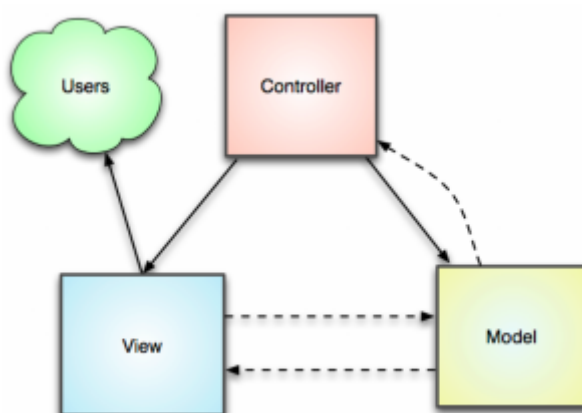


V súčasnosti je jazyk PHP najpoužívanejším jazykom pri tvorbe webových aplikácií. Pre zjednodušenie vývoja webových aplikácií vzniklo množstvo frameworkov, ktoré uľahčujú tvorbu webových aplikácií a určujú akýsi štandard v programovaní. PHP frameworkov existuje pomerne veľa, jeden z najlepších a najpoužívanejších je Zend framework.

Zend Framework je open - source PHP framework používaný pre tvorbu webových aplikácií v jazyku PHP 5. Zend framework je implementovaný s využitím 100% objektovo - orientovaného kódu. Komponentovo založená štruktúra Zend Frameworku je taktiež zaujímavá. Každý komponent je navrhnutý s malými závislosťami na iných komponentoch. Takto jednoducho prepojená architektúra umožňuje použitie jednotlivých komponentov individuálne.

Zend framework je založený na MVC - Model View Controller. MVC je softvérová architektúra, ktorá rozdeľuje dátový model aplikácie, užívateľské rozhranie a riadiacu logiku do troch nezávislých komponentov tak, že modifikácia niektorej z nich má minimálny vplyv na ostatné. Architektúra MVC sa skladá z 3 častí:

- Model - doménovo špecifická reprezentácia informácií, s ktorými aplikácia pracuje
- View (pohľad) prevádza dáta z modelu do podoby vhodnej k prezentácii užívateľovi
- Controller (radič) reaguje na udalosti a zisťuje zmeny v modeli a pohľade



Obrázok 1 : koncept MVC

### Požiadavky

Aktuálna verzia Zend Framework 1.11.11 má nasledovné požiadavky:

- PHP vo verzii 5.2.4. a vyššie
- Server so zapnutým mod\_rewrite.

Takisto je potrebné mať zapnutú direktívu

AllowOverride All

## Inštalácia

Najnovšia verzia Zend frameworku sa dá stiahnuť na [www.zend.com](http://www.zend.com). Pre stiahnutie sa ale musíte zadarmo zaregistrovať. K dispozícii je plná verzia s kompletnou dokumentáciou aj nejakými príkladmi, ale aj minimalizovaná verzia. Odporúča sa použiť plnú verziu. Sú 2 verzie frameworku – pre Windows a Linux. Tento článok sa venuje práve verzii pre Windows s použitím LAMP riešenia, konkrétne XAMPP.

Po rozbalení archívu nájdeme v bin adresári súbor zf.bat. Je to inštalačný súbor Zend frameworku pre Windows. Nasledovným príkazom nainštalujeme Zend framework do adresára zend:

```
#zf.bat create project zend
```

Musíme mať ale nastavený adresár bin ako PATH. Takisto musí byť v PATH aj cesta k php.exe. Potom ešte skopírujeme alebo nalinkujeme adresár library a máme vytvorenú nasledovnú adresárovú štruktúru:

```
|-- application
| |-- Bootstrap.php
| |-- configs
| | `-- application.ini
| |-- controllers
| | |-- ErrorController.php
| | `-- IndexController.php
| |-- models
| `-- views
| |-- helpers
| `-- scripts
| |-- error
| | `-- error.phtml
| `-- index
| `-- index.phtml
|-- library
|-- public
| |-- .htaccess
| `-- index.php
`-- tests
|-- application
| `-- bootstrap.php
|-- library
| `-- bootstrap.php
`-- phpunit.xml
```

Zaujímajú nás nasledovné adresáre:

- application - V tejto zložke je umiestnený celý kód, nevyhnutný na chod aplikácie. Obsahuje všetky zložky MVC patternu - model, view a controller
- library - obsahuje všetky knižnice, potrebné na chod zend frameworku
- public - adresár, kde sú uložené verejne dostupné súbory a to obrázky, javascripty, CSS súbory a iné dokumenty, ku ktorým je nevyhnutný priamy prístup.

Nastavenie Virtual Host:

V nastavení Apache servera v súbore httpd.conf alebo httpd-vhosts.conf musíme nastaviť virtual host a nastaviť ho na priečinok public v adresári, do ktorého sme nainštalovali Zend framework. Urobíme to nasledovne:

```
<VirtualHost *:80>
  ServerName quickstart.local
  DocumentRoot C:\web\zend\public

  SetEnv APPLICATION_ENV "development"

  <Directory C:\web\zend\public>
    DirectoryIndex index.php
    AllowOverride All
    Order allow,deny
    Allow from all
  </Directory>
</VirtualHost>
```

## Controller

Každý controller v zend frameworku by mal byť umiestnený v adresári application/controller. Obsahom súboru je trieda, ktorej názov nesie názov controllera a končí na Controller. Napríklad IndexController. Táto trieda potom obsahuje metódy s názvami akcií, ktoré končia na Action. Napríklad v našom controlleri index je akcia hello, metóda sa teda volá helloAction(). Keď chceme následne zavolať akciu nejakého controllera, musíme zavolať url v tvare /controller/action. To zavolá požadovanú akciu požadovaného controllera.

Príklad application/controllers/IndexController.php:

```
class IndexController extends Zend_Controller_Action {

  public function init()
  {
    /* Initialize action controller here */
  }

  public function indexAction()
  {
    // action body
  }
}
```

```

}
public function helloAction() {

    $hello = new Application_Model_Hello();
    $this->view->result = $hello->getResult();
}
}

```

V našom príklade v metóde helloAction() používame model Hello. Metóda getResult() nám vráti výsledok, ktorý môže model získať napríklad z databázy. Následne priradíme tento výsledok do viewu do premennej result.

## Model

Modely v Zend frameworku umiestňujeme do adresára application/models. Názov súboru je rovnaký ako názov modelu. Model musí obsahovať triedu s názvom v tvare Application\_Model\_NazovModelu. Ako vidíme, názov je poskladaný podľa adresárovej cesty k modelu. Týmto zabezpečíme, že pri volaní modelu je daný súbor automaticky pripojený k aplikácii a nemusíme pri každom použití pripájať súbor cez require\_once(). Toto pripájanie nám zabezpečuje Zend framework. V našom príklade si vytvoríme model Hello, ktorý bude simulovať prácu s databázou a teda nám vráti reťazec Hello world. Výsledok sme použili v controlleri v predchádzajúcom príklade.

Príklad modelu application/models/Hello.php:

```

class Application_Model_Hello {
    public function getResult() {
        //simulation of getting result from DB
        $result = 'Hello world';
        return $result;
    }
}

```

## View

View v Zend frameworku je napísaný v obyčajnom natívnom PHP. Viewy sa umiestňujú do adresára application/views/scripts. Názov view je defaulte rovnaký ako názov akcie daného controllera a umiestňuje sa do adresára s názvom daného controllera. Viewy sa dajú v controlleri podľa potreby meniť, v čom je čaro MVC architektúry. Koncovka každého view musí byť .phtml. Teda view pre akciu hello sa bude volať hello.phtml.

Príklad viewu application/view/scripts/index/hello.phtml

```

<style>
</style>
<div>
This is example output of index controller, hello action and
hello view.<br />
Output from model: <?php echo $this->result; ?>
</div>

```

---

Ak chceme vo viewe použiť nejakú premennú, ktorú sme priradili v controlleri, dostaneme sa k nej cez `$this->result`, kde `$result` je požadovaná premenná z controllera.

Teraz už len stačí zadať do prehliadača url `http://localhost/index/hello` a naša prvá aplikácia v zend frameworku je na svete.

### **Záver:**

Ukázali sme si, že Zend framework je silným nástrojom pre tvorbu webových aplikácií. Inštalácia je pomerne jednoduchá a kto rozumie princípu MVC, nebude mať problém s tvorbou aplikácie v tomto frameworku. Okrem základnej funkcionality, ktorú sme si ukázali v tomto článku, obsahuje množstvo ďalších veľmi silných nástrojov. Na tomto frameworku beží množstvo rozsiahlych aplikácií, čo len svedčí o jeho kvalite.

### **Literatúra**

1. [www.zend.com](http://www.zend.com)
-