

Introdução

Como a idéia deste livro é ir direto ao ponto, vou fazer isso já na introdução.

A idéia desse livro não é explicar a teoria e filosofia do PHP, da orientação a objetos, as maravilhas dos design patterns, etc. Existem ótimos livros e sites que podem lhe ajudar a entender todos os conceitos envolvidos aqui. Entre os livros eu posso indicar:

- PHP Profissional. Alexandre Altair de Melo / Mauricio G. F. Nascimento. Editora Novatec
- PHP Programando com Orientação a Objetos. Pablo Dall'Oglio. Editora Novatec
- Zend Framework Componentes Poderosos para PHP. Flávio Gomes da Silva Lisboa. Editora Novatec
- Zend Framework em Ação. Rob Allen, Nick Lo, Steven Brown. Editora Alta Books.

Os três primeiros são escritos por autores brasileiros e são livros de grande importância e didática. O último é um clássico e também muito bom.

O foco deste livro é ser um guia de desenvolvimento das principais funcionalidades do Zend Framework. Ele iniciou como uma apostila para cursos que venho ministrando nos últimos três ou quatro anos, então é algo que venho testando e alterando continuamente.

Espero que lhe seja útil como tem sido para mim.

Instalando o Zend Framework

Instalar o Zend Framework é uma tarefa simples.

O primeiro passo é verificar os seus requisitos básicos: um servidor web com suporte a reescrita de URLs (Apache será usado nestes exemplos) e o PHP 5.2.4 ou superior.

No arquivo de configuração do Apache basta adicionar as linhas abaixo, ou alterá-las para refletir o seguinte:

```
LoadModule rewrite_module modules/mod_rewrite.so
AddModule mod_rewrite.c
AllowOverride all
```

<https://gist.github.com/987319>

Isto indica ao servidor que ele deve carregar o módulo que permite a reescrita de URLs (mais exemplos nos próximos tópicos) e permite o uso de configurações em arquivos especiais (o .htaccess que vamos ver mais tarde).

Agora basta fazer o download do framework, no site

<http://framework.zend.com>

No momento da escrita deste livro a versão mais atual era a 1.11.6. No site é possível escolher entre três opções para download: a versão com o Zend Server, a versão Full e a versão Minimal do framework. A primeira é indicada se você quer a solução completa, com um servidor Apache e o MySQL já configurados. A versão full possui, além do framework, documentação, testes e demos. E a versão minimal é formada apenas pelo framework. Geralmente a versão minimal é a mais indicada.

Depois de descompactar o arquivo é possível visualizar a seguinte estrutura (para a versão Minimal):

```
bin/ - scripts para a criação de projetos
LICENSE.txt - uma cópia da licença usada pelo framework
README.txt - instruções sobre instalação e documentação
VERSION.txt - texto sobre a versão do framework
library/ - neste diretório encontra-se o framework
```

INSTALL.txt - instruções de instalação

O diretório library deve ser copiado para o diretório htdocs de seu servidor Apache. E pronto! O Zend Framework está pronto para uso.

Definindo o projeto

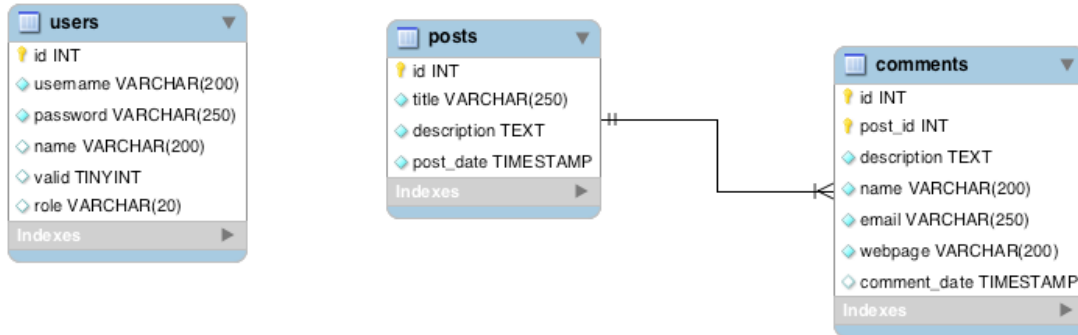
Na minha opinião a única forma de aprender uma nova ferramenta, linguagem, sistema operacional, é quando você realmente precisa resolver algum problema com ela. Pensando nisso, esse livro é baseado na construção de um aplicativo: um blog.

Mas um blog? Por alguns motivos:

- é um problema fácil de se entender. Todo mundo sabe como um blog funciona, seus requisitos e funcionalidades. Então a fase de requisitos do projeto é fácil de completar
- um blog apresenta um grande número de funcionalidades comuns a vários outros sites, como módulos, controle de acesso e permissões, upload de arquivos, tratamento de formulários, cache, traduções, integração com serviços externos, etc.
- a grande maioria dos frameworks possui um exemplo “como desenvolver um blog usando X”, então fica mais fácil para comparação se você já estudou algum outro framework como CakePHP, CodeIgniter ou mesmo Ruby on Rails

Modelagem

Agora que o convenci (ou não) de como desenvolver um blog pode lhe ajudar a entender o Zend Framework, vamos mostrar a modelagem das tabelas:



Simples, como deveria ser.

Usando alguma ferramenta, como o PHPMyAdmin, SequelPro, ou o bom e velho terminal, é possível criar a estrutura do banco usando os comandos SQL abaixo:

```
CREATE DATABASE blog;
```

```
GRANT ALL privileges ON blog.* TO zend@localhost IDENTIFIED BY 'zend';
```

```
USE blog;
```

```
CREATE TABLE IF NOT EXISTS `users` (
  `id` INT NOT NULL AUTO_INCREMENT ,
  `username` VARCHAR(200) NOT NULL ,
  `password` VARCHAR(250) NOT NULL ,
  `name` VARCHAR(200) NULL ,
  `valid` TINYINT NULL ,
  `role` VARCHAR(20) NULL ,
  PRIMARY KEY (`id`) )
ENGINE = InnoDB;
```

```
CREATE TABLE IF NOT EXISTS `posts` (
  `id` INT NOT NULL AUTO_INCREMENT ,
  `title` VARCHAR(250) NOT NULL ,
  `description` TEXT NOT NULL ,
  `post_date` TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP ,
  PRIMARY KEY (`id`) )
ENGINE = InnoDB;
```

```
CREATE TABLE IF NOT EXISTS `comments` (
  `id` INT NOT NULL AUTO_INCREMENT ,
  `post_id` INT NOT NULL ,
  `description` TEXT NOT NULL ,
  `name` VARCHAR(200) NOT NULL ,
  `email` VARCHAR(250) NOT NULL ,
```

```
`webpage` VARCHAR(200) NOT NULL ,
`comment_date` TIMESTAMP NULL ,
PRIMARY KEY (`id`, `post_id`) ,
INDEX `fk_comments_posts` (`post_id` ASC) ,
CONSTRAINT `fk_comments_posts`
  FOREIGN KEY (`post_id` )
  REFERENCES `posts` (`id` )
  ON DELETE NO ACTION
  ON UPDATE NO ACTION)
ENGINE = InnoDB;
```

<https://gist.github.com/987325>

Estrutura do projeto

Uma das vantagens do Zend Framework é sua flexibilidade na hora de montar a estrutura do projeto. A Zend sugere uma estrutura, mas não lhe impede nem dificulta a customização para cada projeto. Nos últimos anos eu venho usando uma estrutura que eu entendo ser mais simples de entender, manter e principalmente apresentar a novos programadores. A estrutura básica é a seguinte:

- controllers/ - diretório onde serão criados os controladores da aplicação
- models/ - diretório onde serão criados os modelos da aplicação
- views/ - diretório onde serão criados os arquivos de visão
- config/config.ini - arquivo de configurações da aplicação, como usuário da base de dados etc.
- public/ - diretório com arquivos que podem ser acessados pelo navegador
 - img/ - diretório com as imagens da aplicação
 - js/ - diretório para os arquivos com scripts JavaScript
 - css/ - diretório para armazenar os arquivos com definição CSS
- index.php - arquivo inicial do projeto com todas as definições da aplicação (também conhecido como bootstrap)
- .htaccess - arquivo com configurações especiais para este diretório

Esta é a estrutura básica para usarmos na grande maioria dos projetos. Neste nosso exemplo, do blog, iremos usar uma estrutura um pouco mais complexa, usando um recurso muito interessante do Zend Framework, que são os módulos. Os módulos foram criados para oferecer um nível maior de separação