



(<http://github.com/oscardias>)



(<https://twitter.com/oscardias>)



(<https://facebook.com/oscardias>)



(<http://google.com/+OscarDias>)



(<http://linkedin.com/company/oscardias>)



(<http://oscardias.com>)

Search



Home (<http://oscardias.com/br/>)

Projetos ▾

Sobre (<http://oscardias.com/br/sobre/>)

CodeIgniter (<http://oscardias.com/br/category/desenvolvimento/php/codeigniter/>)



(http://oscardias.com/br/wp-content/uploads/2012/05/thumb_codeigniter.png)

Criando um App usando CodeIgniter (Parte 1): Introdução

👤 Oscar Dias (<http://oscardias.com/br/author/oscardias/>) 📅 16 de maio de 2012 👁 Criando um App usando CodeIgniter ▾ 2 Comments
(http://oscardias.com/br/desenvolvimento/php/codeigniter/criando-um-app-usando-codeigniter-parte-1-introducao/#disqus_thread)

Semana passada (<http://oscardias.com/br/desenvolvimento/php/codeigniter/projeto-em-php-simple-task-board/>) eu escrevi sobre um pequeno aplicativo desenvolvido utilizando CodeIgniter chamado Simple Task Board. Ele é capaz de gerenciar projetos e tarefas, sendo muito simples, especialmente agora que está na primeira versão. Então achei que seria uma boa ideia explicar como ele foi feito. Essa explicação vai se estender por alguns artigos, começando hoje com uma introdução sobre CodeIgniter e sobre a aplicação em si.

Introdução

CodeIgniter (<http://codeigniter.com/>) é um Framework MVC (model-view-controller, ou modelo-visão-controlador) para PHP muito level. Isso significa que ele possui diversas funções que ajudam a organizar e acelerar o processo de desenvolvimento de aplicações web. Além disso ele é bem simples quando comparado com outros frameworks.

A comunidade envolvida com um framework também deve ser levada em consideração ao se escolher. CodeIgniter tem uma comunidade bem ativa (em inglês e em português) e um bom guia de usuário, então é fácil encontrar a informação que precisar. Vou explicar o que MVC significa.

Model-View-Controller

Traduzindo, ficaria Modelo-Visão-Controlador). É um design pattern que divide a aplicação em três partes:

- Modelo: interação com o banco de dados e lógica de negócio. Operações CRUD (create, read, update, delete).
- Visão: o que o usuário vê. Nosso HTML.
- Controlador: lógica da aplicação e tratamento de input. O controlador utiliza o modelo para buscar/registrar informações no banco de dados e chama as visões necessárias para exibir para o usuário.

Numa explicação simplória, é uma arquitetura ou modelo para estruturar sua aplicação. Para quem já trabalhou com aplicações em PHP puro que tratam inputs no mesmo arquivo que possui o HTML, incluem conexões com banco de dados e executam operações de banco, trabalhar com um framework MVC é extremamente bom. No início pode parecer complicado, mas com um pouquinho de tempo para entender a simplicidade e organização do código, você não vai mais trabalhar sem ele.

Estrutura e Arquivos

A instalação do CodeIgniter tem uma estrutura padrão. Isso pode ser alterado mudando alguns arquivos, mas para esse aplicativo vamos manter os diretórios como estão. Então vá para <http://www.codeigniter.com> (<http://www.codeigniter.com>) e baixe a última versão (atualmente 2.1.0). Descomprima o conteúdo na pasta que você escolher. Isso é o que você vai ver:

- application
- system
- user_guide
- index.php
- license.txt

O arquivo *index.php* indica o ambiente (*desenvolvimento, produção, etc*) e as pastas de sistema e aplicação. Então, se você quiser, você pode renomear essas pastas para outra coisa e modificar esses valores no arquivo *index.php*. Isso ajuda na segurança do seu aplicativo.

O arquivo *license.txt* contém o acordo de uso do CodeIgniter.

application

Nesta pasta ficam todas as informações específicas da aplicação. As subpastas mais importantes, onde você vai trabalhar mais, são:

- config: configuração do app. Dados de acesso ao banco de dados, auto carregamento de bibliotecas e helpers, rotas, etc.
- controllers: controladores do app. Cada controlador é uma classe PHP que estende a classe *CI_Controller*. Para acessar o controlador é só você acessar *server/controller_name* no seu navegador.
- models: modelos para acessar o banco de dados. Os modelos também são classes, mas elas estendem a classe *CI_Model*.
- views: todas as suas visões estarão aqui.

Existem várias outras subpastas, como *libraries* (bibliotecas) e *helpers*, onde você pode adicionar suas classes específicas e suas funções.

system

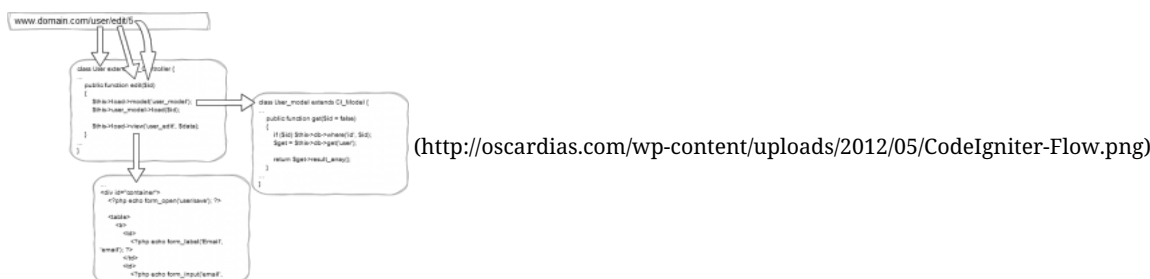
Este diretório contém a “engine” do CodeIgniter. Todas as bibliotecas do sistema estão aqui, então você não precisa se preocupar com o que está aqui..

user_guide

Como você deve estar imaginando, nesta pasta está o guia do usuário que acompanha o CodeIgniter. Você pode remover esta pasta ou manter ela em um local separa para referências futuras. Você pode visitar este guia do usuário online (http://codeigniter.com/user_guide/) também.

Fluxo Simplificado

Este é um fluxo simplificado para explicar como um cenário simples funciona.



Nesse exemplo nós estamos acessando um usuário para editar seus detalhes. O CodeIgniter trata as URLs de uma maneira simples, utilizando a primeira parte como o nome do controlador e a próxima como o método. Outras partes podem ser parâmetros do método. Se o controlador não for definido, o controlador padrão será executado (isso é definido no diretório *config*). Se o método não estiver definido, o método *index* será utilizado por padrão.

Eu vou explicar em detalhe como nós fazemos cada coisa nos próximos artigos. O meu objetivo é ir passo a passo, explicando como o CodeIgniter funciona em situações práticas. Agora deixe-me explicar o que será o aplicativo que estaremos desenvolvendo.

Simple Task Board

O aplicativo que estaremos criando é o Simple Task Board (traduzindo, Quadro de Tarefas Simples). Ele está disponível aqui neste blog (<http://oscardias.com/br/desenvolvimento/php/codeigniter/projeto-em-php-simple-task-board/>) e também no GitHub (<https://github.com/oscardias/Simple-Task-Board>). Foi criado por mim mesmo, para me ajudar a organizar meus projetos pessoais. Eu decidi disponibilizá-lo aqui para poder fazer melhorias e, talvez, usar ele para gerenciar os projetos na minha empresa. Como eu não preciso e não quero perder muito tempo gerenciando os projetos, eu não preciso de muitas funcionalidades. Eu acabo usando um *agile board* ou *simples planilhas*. Por isso o Simple Task Board poderia ser customizado para compreender apenas as necessidades mais básicas, sendo ao mesmo tempo simples e útil.

Conclusão

Isso é tudo para a Introdução. Já fiz uma nova instalação do CodeIgniter na minha máquina local, então estou pronto para dar início no tutorial passo a passo. Na próxima semana vou preparar a parte 2 do tutorial. Tenha um bom resto de semana.

CodeIgniter (<http://oscardias.com/br/tag/codeigniter/>) PHP (<http://oscardias.com/br/tag/php/>) Quadro de Tarefas (<http://oscardias.com/br/tag/quadro-de-tarefas/>)

- Este post é parte da série Criando um App usando CodeIgniter (<http://oscardias.com/br/series/criando-um-app-usando-codeigniter/>).
- Criando um App usando CodeIgniter (Parte 1): Introdução
 - Criando um App usando CodeIgniter (Parte 2): Instalação (<http://oscardias.com/br/desenvolvimento/php/codeigniter/criando-um-app-usando-codeigniter-parte-2-instalacao/>)
 - Criando um App usando CodeIgniter (Parte 3): Login (<http://oscardias.com/br/desenvolvimento/php/codeigniter/criando-um-app-usando-codeigniter-parte-3-login/>)
 - Criando um App usando CodeIgniter (Parte 4): Usuários (<http://oscardias.com/br/desenvolvimento/php/codeigniter/criando-um-app-usando-codeigniter-parte-4-usuarios/>)
 - Criando um App usando CodeIgniter (Parte 5): Projetos (<http://oscardias.com/br/desenvolvimento/php/codeigniter/criando-um-app-usando-codeigniter-parte-5-projetos/>)
 - Criando um App usando CodeIgniter (Parte 6): Tarefas (<http://oscardias.com/br/desenvolvimento/php/codeigniter/criando-um-app-usando-codeigniter-parte-6-tarefas/>)
 - Criando um App usando CodeIgniter (Parte 7): Validações (<http://oscardias.com/br/desenvolvimento/php/codeigniter/criando-um-app-usando-codeigniter-parte-7-validacoes/>)

Share

 (<https://twitter.com/oscardias>)

 (<https://www.facebook.com/oscardias/>)

 (<http://www.linkedin.com/company/oscardias/>)

 (<https://plus.google.com/u/0/+OscarDias1984/>)

 (<http://www.stumbleupon.com/author/oscardias/>)

Sobre Oscar Dias (<http://oscardias.com/br/author/oscardias/>)



Trabalho com desenvolvimento desde 2003 e já utilizei diversas linguagens de programação. Atualmente trabalho com PHP na minha empresa Softerize Sistemas.

Sort by Best ▾

Share  Favorite ★

Join the discussion...



Rafael

a year ago



Oscar, sou programador desktop e nunca havia programado para web antes, de uns meses pra cá estou a procura de algum tutorial ou algo parecido para web (acho que há muitas pessoas na mesma situação), mas não havia achado nada bem completo até agora. Recomendo... Principalmente pela didática do tutorial e pelo conteúdo. Ótimo trabalho e iniciativa... obrigado!

0 ^ ▾ Reply



Ribamar FS

2 years ago



Um excelente trabalho.
Muito obrigado!

0 ^ ▾ Reply

 Subscribe

 Add Disqus to your site

 Privacy