

OpenCart: como fizemos
um módulo diferente de
tudo o que se vê por aí



Quem sou eu?



Michel Lima

- Cientista da computação e programador PHP fullstack há 7 anos.
- Especializado em e-commerce
- Desenvolvedor na Mundipagg



Agenda

- Opencart como plataforma
- Vantagens e desvantagens
- A guerra dos clones
- O desafio
- O início
- Desacoplando
- Ferramentas
- O que vem pela frente?
- Dúvidas
- Agradecimento

NAVIGATION

Dashboard

Catalog

Extensions

Design

Sales

Customers

Marketing

System

Reports

Orders Completed 0%

Orders Processing 0%

Other Statuses 0%

Dashboard

Home > Dashboard

TOTAL ORDERS 0%



0

View more...

TOTAL SALES 0%



0

View more...

TOTAL CUSTOMERS 0%



0

View more...

PEOPLE ONLINE



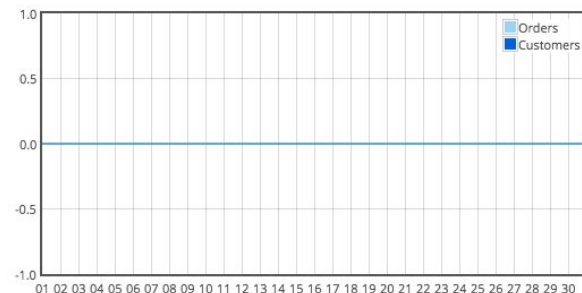
0

View more...

World Map



Sales Analytics



Recent Activity

No results!

Latest Orders

Order ID	Customer	Status	Date Added	Total	Action
No results!					



Opencart como plataforma

- Simplicidade como lema



Vantagens

- Fácil instalação (suporte a cli)
- Curva de aprendizado acentuada
- Leve e rápido
- Admin intuitivo
- Marketplace com milhares de módulos
- Pacote com 15mb (já com sample data)



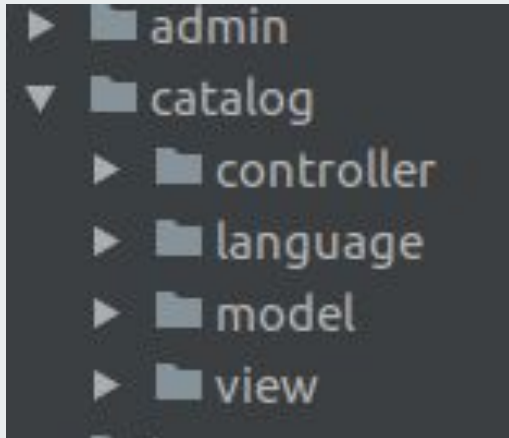
Desvantagens

- Sem suporte a Composer
- Padrão próprio
- Documentação pobre



Opencart como plataforma

- MVCL simples



A guerra dos clones



```
public function index() {  
    $this->load->language('extension/payment/amazon_login_pay');  
    $this->document->setTitle($this->language->get('heading_title'));  
    $this->load->model('setting/setting');  
    $this->load->model('extension/payment/amazon_login_pay');  
    if (($this->request->server['REQUEST_METHOD'] == 'POST') && $this->validate()) {  
        $this->model_setting_setting->editSetting('payment_amazon_login_pay', $this->request->post);  
        $this->session->data['success'] = $this->language->get('text_success');  
        if (isset($this->request->post['language_reload'])) {  
            $this->response->redirect($this->url->link('payment/amazon_login_pay', 'user_token=' . $this->session->data['user_token'], true));  
        } else {  
            $this->response->redirect($this->url->link('marketplace/extension', 'user_token=' . $this->session->data['user_token'] . '&type=payment', true));  
        }  
    }  
    if (isset($this->error['warning'])) {  
        $data['error_warning'] = $this->error['warning'];  
    } else {  
        $data['error_warning'] = '';  
    }  
    if (isset($this->error['error_merchant_id'])) {  
        $data['error_merchant_id'] = $this->error['error_merchant_id'];  
    } else {  
        $data['error_merchant_id'] = '';  
    }  
    if (isset($this->error['error_access_key'])) {  
        $data['error_access_key'] = $this->error['error_access_key'];  
    } else {  
        $data['error_access_key'] = '';  
    }  
    if (isset($this->error['error_access_secret'])) {  
        $data['error_access_secret'] = $this->error['error_access_secret'];  
    } else {  
        $data['error_access_secret'] = '';  
    }  
}
```

```

public function index() {
    $this->load->language('extension/payment/alipay');

    $this->document->setTitle($this->language->get('heading_title'));

    $this->load->model('setting/setting');

    if (($this->request->server['REQUEST_METHOD'] == 'POST') && $this->validate()) {
        $this->model_setting->editSetting('payment_alipay', $this->request->post);

        $this->session->data['success'] = $this->language->get('text_success');

        $this->response->redirect($this->url->link('marketplace/extension', 'user_token=' . $this->session->data['user_token'] . '&type=payment', true));
    }

    if (isset($this->error['warning'])) {
        $data['error_warning'] = $this->error['warning'];
    } else {
        $data['error_warning'] = '';
    }

    if (isset($this->error['app_id'])) {
        $data['error_app_id'] = $this->error['app_id'];
    } else {
        $data['error_app_id'] = '';
    }

    if (isset($this->error['merchant_private_key'])) {
        $data['error_merchant_private_key'] = $this->error['merchant_private_key'];
    } else {
        $data['error_merchant_private_key'] = '';
    }

    if (isset($this->error['alipay_public_key'])) {
        $data['error_alipay_public_key'] = $this->error['alipay_public_key'];
    } else {
        $data['error_alipay_public_key'] = '';
    }

    $data['breadcrumbs'] = array();

    $data['breadcrumbs'][] = array(
        'text' => $this->language->get('text_home'),
        'href' => $this->url->link('common/home', 'user_token=' . $this->session->data['user_token'] . '&type=payment', true)
    );
}

```

```
public function index() {
    $this->load->language('extension/payment/pp_express');

    $this->document->setTitle($this->language->get('heading_title'));

    $this->load->model('setting/setting');

    if (($this->request->server['REQUEST_METHOD'] == 'POST') && $this->validate()) {
        $this->model_setting_setting->editSetting('payment_pp_express', $this->request->post);

        $this->session->data['success'] = $this->language->get('text_success');

        $this->response->redirect($this->url->link('marketplace/extension', 'user_token=' . $this->session->data['user_token'] . '&type=payment', true));
    }

    $data['user_token'] = $this->session->data['user_token'];

    $data['button_configure'] = $this->url->link('extension/module/pp_button/configure', 'user_token=' . $this->session->data['user_token'], true);

    if (isset($this->error['warning'])) {
        $data['error_warning'] = $this->error['warning'];
    } else {
        $data['error_warning'] = '';
    }

    if (isset($this->error['username'])) {
        $data['error_username'] = $this->error['username'];
    } else {
        $data['error_username'] = '';
    }

    if (isset($this->error['password'])) {
        $data['error_password'] = $this->error['password'];
    } else {
        $data['error_password'] = '';
    }

    if (isset($this->error['signature'])) {
        $data['error_signature'] = $this->error['signature'];
    } else {
        $data['error_signature'] = '';
    }

    if (isset($this->error['sandbox_username'])) {
        $data['error_sandbox_username'] = $this->error['sandbox_username'];
    }
}
```

```
public function index() {
    $this->load->language('extension/payment/globalpay');

    $this->document->setTitle($this->language->get('heading_title'));

    $this->load->model('setting/setting');

    if ((($this->request->server['REQUEST_METHOD'] == 'POST') && $this->validate()) {
        $this->model_setting_setting->editSetting('payment_globalpay', $this->request->post);

        $this->session->data['success'] = $this->language->get('text_success');

        $this->response->redirect($this->url->link('marketplace/extension', 'user_token=' . $this->session->data['user_token'] . '&type=payment', true));
    }

    $data['notify_url'] = HTTPS_CATALOG . 'index.php?route=extension/payment/globalpay/notify';

    if (isset($this->error['warning'])) {
        $data['error_warning'] = $this->error['warning'];
    } else {
        $data['error_warning'] = '';
    }

    if (isset($this->error['error_merchant_id'])) {
        $data['error_merchant_id'] = $this->error['error_merchant_id'];
    } else {
        $data['error_merchant_id'] = '';
    }

    if (isset($this->error['error_secret'])) {
        $data['error_secret'] = $this->error['error_secret'];
    } else {
        $data['error_secret'] = '';
    }

    if (isset($this->error['error_live_url'])) {
        $data['error_live_url'] = $this->error['error_live_url'];
    } else {
        $data['error_live_url'] = '';
    }

    if (isset($this->error['error_demo_url'])) {
        $data['error_demo_url'] = $this->error['error_demo_url'];
    } else {
        $data['error_demo_url'] = '';
    }
}
```

```
public function index() {
    $this->load->language('extension/payment/wechat_pay');

    $this->document->setTitle($this->language->get('heading_title'));

    $this->load->model('setting/setting');

    if (($this->request->server['REQUEST_METHOD'] == 'POST') && $this->validate()) {
        $this->model_setting_setting->editSetting('payment_wechat_pay', $this->request->post);

        $this->session->data['success'] = $this->language->get('text_success');

        $this->response->redirect($this->url->link('marketplace/extension', 'user_token=' . $this->session->data['user_token'] . '&type=payment', true));
    }

    if (isset($this->error['warning'])) {
        $data['error_warning'] = $this->error['warning'];
    } else {
        $data['error_warning'] = '';
    }

    if (isset($this->error['app_id'])) {
        $data['error_app_id'] = $this->error['app_id'];
    } else {
        $data['error_app_id'] = '';
    }

    if (isset($this->error['app_secret'])) {
        $data['error_app_secret'] = $this->error['app_secret'];
    } else {
        $data['error_app_secret'] = '';
    }

    if (isset($this->error['mch_id'])) {
        $data['error_mch_id'] = $this->error['mch_id'];
    } else {
        $data['error_mch_id'] = '';
    }

    if (isset($this->error['api_secret'])) {
        $data['error_api_secret'] = $this->error['api_secret'];
    } else {
        $data['error_api_secret'] = '';
    }
}
```


Cheque

```
public function index() {
    $this->load->language('extension/payment/cheque');

    $this->document->setTitle($this->language->get('heading_title'));

    $this->load->model('setting/setting');

    if (($this->request->server['REQUEST_METHOD'] == 'POST') && $this->validate()) {
        $this->model_setting_setting->editSetting('payment_cheque', $this->request->post);

        $this->session->data['success'] = $this->language->get('text_success');

        $this->response->redirect($this->url->link('marketplace/extension', 'user_token=' . $this->session->data['user_token'] . '&type=payment', true));
    }

    if (isset($this->error['warning'])) {
        $data['error_warning'] = $this->error['warning'];
    } else {
        $data['error_warning'] = '';
    }

    if (isset($this->error['payable'])) {
        $data['error_payable'] = $this->error['payable'];
    } else {
        $data['error_payable'] = '';
    }

    $data['breadcrumbs'] = array();

    $data['breadcrumbs'][] = array(
        'text' => $this->language->get('text_home'),
        'href' => $this->url->link('common/dashboard', 'user_token=' . $this->session->data['user_token'], true)
    );

    $data['breadcrumbs'][] = array(
        'text' => $this->language->get('text_extension'),
        'href' => $this->url->link('marketplace/extension', 'user_token=' . $this->session->data['user_token'] . '&type=payment', true)
    );

    $data['breadcrumbs'][] = array(
        'text' => $this->language->get('heading_title'),
        'href' => $this->url->link('extension/payment/cheque', 'user_token=' . $this->session->data['user_token'], true)
    );

    $data['action'] = $this->url->link('extension/payment/cheque', 'user_token=' . $this->session->data['user_token'], true);
}
```



O desafio

Construir um módulo de integração com nossa API que possua diversos métodos de pagamento, atualização síncrona e assíncrona de status de pedidos, compra com um clique, sem passar os dados sensíveis de cartão de crédito pelo backend (requisito PCI), forneça planos e assinaturas e tudo isso sem ser mais uma cópia dos outros módulos.



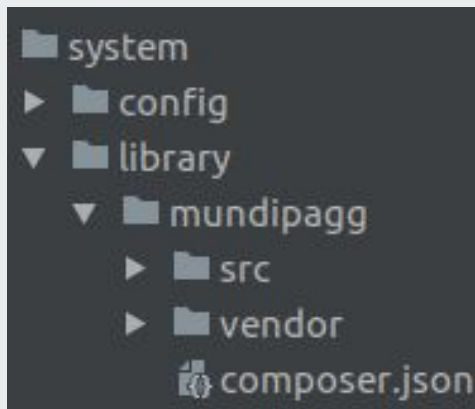
O início

O que queríamos:

- 1 - Utilizar nossa sdk dentro de um projeto sem composer e sem namespace.
- 2 - Deixar nosso código dentro das psr's e padrões atuais do php.
- 3 - Usar namespace, o autoload do composer, aumentar a testabilidade do código (fruto do isolamento).



Desacoplando



Trouxemos a maior parte do nosso código para uma única pasta dentro de system/library, concentrando tudo em um único lugar, tornando possível a utilização de nossa sdk e do composer



Desacoplando

```
namespace Mundipagg\Controller;  
  
use Mundipagg\Model\Creditcard;  
  
class SavedCreditCard  
{  
    private $openCart;  
  
    public function __construct($openCart)  
    {  
        $this->openCart = $openCart;  
    }  
}
```

Ao instanciar um controller nosso, passamos para seu construtor, o objeto da classe controller do Opencart, tornando possível o acesso a todos os métodos e propriedades originais



O problema Vqmod

Tutorial de como não fazer modificações em seu sistema

```
<file path="system/{engine,library}/{action,loader,config,language}*.php|system/library/template/template.php">
  <operation>
    <search regex="true">
      <![CDATA[~(require|include)( _once)?\(([^\])+~)]>
    </search>
    <add position="replace">
      <![CDATA[$1$2(modification($3))]]>
    </add>
  </operation>
</file>
```



Eventos

```
//Add button to order list in admin  
$this->model_setting_event->addEvent(  
    'payment_mundipagg_add_order_actions',  
    'admin/view/sale/order_list/before',  
    'extension/payment/mundipagg/callEvents'  
);
```

A implementação dos eventos no Opencart 2.x possibilitou a substituição do Vqmod por algo mais confiável.

Fazer alterações não precisa mais ser doloroso.

Resultado

```
/**
 * Load basic data and call postRequest or getRequest methods accordingly
 *
 * @return void
 */
public function index()
{
    $this->load->language('extension/payment/mundipagg');
    $this->document->setTitle($this->language->get('heading_title'));
    $this->load->model('setting/setting');

    if ($this->request->server['REQUEST_METHOD'] == 'POST') {
        $this->postRequest();
    } else {
        $this->getRequest();
    }
}
```



O que vem pela frente?

- Planos e assinaturas
- CI e CD (em andamento)
- Transferir mais código de dentro da estrutura do Opencart para a nossa
- Tornar mais completo e preciso o sistema de logs
- Convergência dos módulos da Mundipagg



Ferramentas

- Modman
- Ngrok
- Mundipagg SDK
- Postman
- Circle CI



Contato



github.com/michelp1



[@Michel_PL](https://twitter.com/Michel_PL)



Útil

- Opencart (<https://www.opencart.com>)
- Módulo Mundipagg (<https://github.com/mundipagg/opencart>)
- Modman (<https://github.com/colinmollenhour/modman>)
- Ngrok (<https://ngrok.com/>)
- Postman (<https://www.getpostman.com/>)
- Circle CI (<https://circleci.com/>)
- Vhost (<https://github.com/felipebool/vhost>)
- FrontLink (<https://github.com/michelp/FrontLink>)