

Integración del SDK de Ping++ con el Framework CodeIgniter

A continuación se presenta un ejemplo completo, paso a paso, de cómo integrar **Pingpp** (Ping++), un SDK de procesamiento de pagos, en una aplicación PHP utilizando el marco de trabajo CodeIgniter, basado en la clase `BaseController` proporcionada. Esta guía asume que tiene una configuración básica de CodeIgniter y se centra en la integración de Pingpp para el procesamiento de pagos, como habilitar a los usuarios para pagar a través de Alipay PC Direct (aunque puede adaptarlo para otros canales).

Requisitos previos

- Cuenta de Pingpp:** Regístrese en Ping++ y obtenga su **API Key** y **App ID**. Tendrá una clave de prueba (`sk_test_...`) para el desarrollo y una clave en vivo (`sk_live_...`) para la producción.
- SDK de PHP de Pingpp:** Instale el SDK de PHP de Pingpp a través de Composer:

```
composer require pingplusplus/pingpp-php
```

Asegúrese de que el SDK se cargue automáticamente en su aplicación CodeIgniter (por ejemplo, colóquelo en `application/libraries` o configure la carga automática de Composer).

- Configuración de CodeIgniter:** Debería tener una aplicación CodeIgniter en funcionamiento con un controlador y modelos de base de datos para usuarios y cargos.
-

Integración Paso a Paso

- Configurar el Controlador** Cree un controlador (por ejemplo, `PaymentController.php`) que extienda un controlador base similar a `BaseController`. Configure la clave API de Pingpp en el constructor según su entorno (desarrollo o producción).

```
<?php  
defined('BASEPATH') OR exit('No direct script access allowed');  
  
require_once APPPATH . '/libraries/REST_Controller.php';  
require_once APPPATH . '/vendor/autoload.php'; // Suponiendo carga automática de Composer  
  
class PaymentController extends REST_Controller {  
  
    public $userDao;
```

```

public $chargeDao;

public function __construct() {
    parent::__construct();

    // Establecer clave API de Pingpp según el entorno
    $isLocalDebug = ENVIRONMENT === 'development'; // Entorno de CodeIgniter
    if ($isLocalDebug) {
        \Pingpp\Pingpp::setApiKey('sk_test_your_test_key_here'); // Reemplace con su clave de prueba
    } else {
        \Pingpp\Pingpp::setApiKey('sk_live_your_live_key_here'); // Reemplace con su clave en vivo
    }

    // Cargar modelos
    $this->load->model('UserDao');
    $this->userDao = new UserDao();
    $this->load->model('ChargeDao');
    $this->chargeDao = new ChargeDao();
}
}

```

- Reemplace 'sk_test_your_test_key_here' y 'sk_live_your_live_key_here' con sus claves API de Pingpp reales.
- Asegúrese de que UserDao y ChargeDao estén implementados para manejar la autenticación de usuarios y el almacenamiento de cargos (consulte el Paso 2).

2. Crear Modelos de Soporte

Necesitará modelos para gestionar sesiones de usuarios y almacenar detalles de cargos.

UserDao.php (ejemplo simplificado):

```

<?php
class UserDao extends CI_Model {
    public function findUserBySessionToken($token) {
        $query = $this->db->get_where('users', ['session_token' => $token]);
        return $query->row(); // Devuelve objeto de usuario o null
    }
}

```

ChargeDao.php (ejemplo simplificado):

```

<?php

class ChargeDao extends CI_Model {

    public function add($orderNo, $amount, $userId, $ipAddress) {
        $data = [
            'order_no' => $orderNo,
            'amount' => $amount,
            'user_id' => $userId,
            'client_ip' => $ipAddress,
            'status' => 'pending',
            'created_at' => date('Y-m-d H:i:s')
        ];
        $this->db->insert('charges', $data);
    }
}

```

Cree tablas de base de datos correspondientes (`users` y `charges`) con campos adecuados.

3. Implementar el Método de Creación de Cargo Agregue un método para crear un cargo de Pingpp y devolverlo al cliente. Este ejemplo utiliza Alipay PC Direct como canal de pago.

```

public function createChargeThenResponse($amount, $subject, $body, $metaData, $user) {
    // Generar un número de pedido único
    $orderNo = $this->generateOrderNo();

    // Establecer App ID según el entorno
    $isLocalDebug = ENVIRONMENT === 'development';
    $appId = $isLocalDebug ? 'app_your_test_app_id' : 'app_your_live_app_id';

    // Obtener dirección IP del cliente
    $ipAddress = $this->input->ip_address();
    if ($ipAddress === '::1') { // Manejar caso de depuración local
        $ipAddress = '127.0.0.1';
    }

    // Crear el cargo
    try {
        $charge = \Pingpp\Charge::create([
            'order_no' => $orderNo,
            'app' => ['id' => $appId],
            'channel' => 'alipay_pc_direct', // Cambie esto para otros canales (por ejemplo, 'wx' para WeChat)
        ]);
    }
}

```

```

'amount' => $amount, // En centavos (por ejemplo, 1000 = 10 CNY)
'client_ip' => $ipAddress,
'currency' => 'cny',
'subject' => $subject,
'body' => $body,
'metadata' => $metaData,
'extra' => [
    'success_url' => 'http://yourdomain.com/payment/success' // Reemplace con su URL de éxito
]
]);

// Almacenar detalles del cargo en la base de datos
$this->chargeDao->add($orderNo, $amount, $user->id, $ipAddress);

// Devolver el objeto de cargo como JSON
$this->output
    ->set_status_header(200)
    ->set_content_type('application/json', 'utf-8')
    ->set_output(json_encode($charge));
} catch (\Pingpp\Error\Base $e) {
    log_message('error', 'Cargo de Pingpp Fallido: ' . $e->getMessage());
    $this->response(['error' => 'Fallo en la creación del pago'], 500);
}
}

private function generateOrderNo() {
    return uniqid(); // ID único simple; reemplace con un generador más robusto si es necesario
}

```

- Reemplace 'app_your_test_app_id' y 'app_your_live_app_id' con sus IDs de App de Pingpp.
- Ajuste 'success_url' a la URL de la página de éxito de su aplicación.

4. Crear un Método de Pago

Agregue un método público para manejar el pago del usuario, llamando al método de creación de cargo.

```

public function checkout_post() {
    // Verificar si el usuario ha iniciado sesión
    $user = $this->getSessionUser();
    if (!$user) {
        $this->response(['error' => 'Usuario no ha iniciado sesión'], 401);
    }
}

```

```

    return;
}

// Detalles de pedido de ejemplo
$amount = 1000; // 10 CNY (ejemplo; calcule dinámicamente en una aplicación real)
$subject = 'Pago de pedido';
$body = 'Pago para el pedido #123';
$metaData = ['order_id' => '123']; // Adjuntar datos específicos del pedido

// Crear cargo y responder
$this->createChargeThenResponse($amount, $subject, $body, $metaData, $user);
}

private function getSessionUser() {
    $token = $this->input->get_request_header('Authorization', TRUE); // Ajuste según su método de autenticación
    if ($token) {
        return $this->userDao->findUserBySessionToken($token);
    }
    return null;
}

```

5. Manejar la Respuesta del Cliente El método `createChargeThenResponse` devuelve un objeto JSON `charge`. Para `alipay_pc_direct`, incluye un campo `credential` con una URL de pago. Maneje esto en el lado del cliente (por ejemplo, con JavaScript).

Ejemplo (JavaScript del Frontend):

```

fetch('/payment/checkout', {
    method: 'POST',
    headers: { 'Authorization': 'your-session-token' }
})
.then(response => response.json())
.then(charge => {
    if (charge.credential && charge.credential.alipay_pc_direct) {
        window.location.href = charge.credential.alipay_pc_direct; // Redirigir a Alipay
    } else {
        console.error('Error de pago:', charge);
    }
})
.catch(error => console.error('Error:', error));

```

6. Página de Éxito Cree una página de éxito (por ejemplo, payment/success) para mostrar una confirmación después del pago. Actualice el estado del pedido manualmente aquí si es necesario, aunque se recomiendan los webhooks para la confiabilidad (consulte el Paso 7).

```
public function success_get() {  
    // Opcionalmente, verifique el estado del pago aquí  
    $this->load->view('payment_success'); // Cargar una vista  
}
```

7. (Opcional) Manejar Webhooks Pingpp envía actualizaciones de estado de pago asíncronas a través de webhooks. Configure un punto final para recibir estas notificaciones y actualice su base de datos.

```
public function webhook_post() {  
    $rawData = file_get_contents('php://input');  
    $event = json_decode($rawData, true);  
  
    if ($event['type'] === 'charge.succeeded') {  
        $charge = $event['data']['object'];  
        $orderNo = $charge['order_no'];  
        $this->chargeDao->updateStatus($orderNo, 'paid');  
    }  
  
    $this->response(['status' => 'ok'], 200);  
}
```

- Agregue un método updateStatus a ChargeDao para actualizar el estado del cargo.
- Configure su URL de webhook en el tablero de Pingpp (por ejemplo, <http://yourdomain.com/payment/webhook>).

Uso Completo del Ejemplo

1. **Acción del Usuario:** Un usuario inicia sesión e inicia un pago mediante una solicitud POST a /payment/checkout.
2. **Respuesta del Servidor:** El servidor crea un cargo y devuelve un objeto JSON con detalles de pago.
3. **Manejo del Cliente:** El frontend redirige al usuario a la página de pago de Alipay.
4. **Completar el Pago:** Despues del pago, el usuario es redirigido a la URL de éxito.
5. **Actualización del Webhook:** Pingpp notifica su punto final de webhook y actualiza el estado del cargo.

Notas Adicionales

- **Canales de Pago:** Reemplace 'alipay_pc_direct' con otros canales (por ejemplo, 'wx' para WeChat) según sea necesario. Consulte la documentación de Pingpp para parámetros extra específicos del canal.
- **Manejo de Errores:** Mejore el manejo de errores para la producción (por ejemplo, lógica de reintento, mensajes de error detallados).
- **Seguridad:** Valide la entrada del usuario (\$amount, etc.) y asegure sus puntos finales de API.

Este ejemplo proporciona una base sólida para integrar Pingpp en su aplicación CodeIgniter, adaptable a diversos casos de uso como comercio electrónico o donaciones.